Package ‘gazepath’

February 9, 2020

Type Package
Title Parse Eye-Tracking Data into Fixations
Version 1.3
Date 2020-02-07
Author Daan van Renswoude & Ingmar Visser
Maintainer Daan van Renswoude <D.R.vanRenswoude@uva.nl>
Description Eye-tracking data must be transformed into fixations and saccades before it can be analyzed. This package provides a non-parametric speed-based approach to do this on a trial basis. The method is especially useful when there are large differences in data quality, as the thresholds are adjusted accordingly. The same pre-processing procedure can be applied to all participants, while accounting for individual differences in data quality.
License GPL-2
Imports sp, jpeg, zoo, scales, shiny
Repository CRAN
Repository/R-Forge/Project gazepatterns
Repository/R-Forge/Revision 116
Repository/R-Forge/DateTimeStamp 2020-02-08 12:40:26
Date/Publication 2020-02-09 15:20:02 UTC
NeedsCompilation no
Depends R (>= 2.10)

R topics documented:

  gazepath-package .................................................. 2
eye_dat .............................................................. 2
gazepath .............................................................. 3
GUI ................................................................. 7
screen ................................................................. 8

Index 9
gazepath-package  Parse Eye-Tracking Data into Fixations

Description

Eye-tracking data must be transformed into fixations and saccades before it can be analyzed. This package provides a non-parametric speed-based approach to do this on a trial basis. The method is especially useful when there are large differences in data quality, as the thresholds are adjusted accordingly. The same pre-processing procedure can be applied to all participants, while accounting for individual differences in data quality.

Details

Package:  gazepath
Type:  Package
Version:  1.21
Date:  2017-05-22
License:  GPL-2

Author(s)

Daan van Renswoude & Ingmar Visser

Maintainer: Daan van Renswoude <D.R.vanRenswoude@uva.nl>

References


data(eye_dat)

Description

eye-tracking data of an 11-month-old infant watching 15 real-world scene stimuli

Usage

data(eye_dat)
**gazepath**

**Format**

A dataframe with 4 variables (x- and y-coordinates of gazeposition in pixels, stimuli name and distance to the screen in mm)

**Source**

S. P. Johnson

**References**

S. P. Johnson

---

### gazepath

**Transform Raw Eye-Tracking Data into Fixations and Saccades**

**Description**

This function transforms raw eye-tracking data into fixations and saccades using a non-parametric speed-based approach to do this on a trial basis. The method is especially useful when there are large differences in data quality between or within participants. Lower data quality results in more conservative thresholds, thereby controlling for noise levels.

**Usage**

```r
gazepath(data, x1, y1, x2 = NULL, y2 = NULL, d1, d2 = NULL, trial, height_px, height_mm, width_px, width_mm, extra_var = NULL, res_x = 1280, res_y = 1024, samplerate = 500, method = "Mould", posthoc = FALSE, thres_vel = 35, thres_dur = 100, min_dist = 250, in_thres = 150)
```

```r
## S3 method for class 'gazepath'
summary(object, ..., complete_only = FALSE, fixations_only = FALSE)
## S3 method for class 'gazepath'
plot(x, ..., trial_index = 1)
## S3 method for class 'gazepath'
print(x, ...)
## S3 method for class 'gazepath'
head(x, ...)
## S3 method for class 'gazepath'
tail(x, ...)
```
Arguments

data
The dataframe with at least the raw x- and y-coordinates, the distance to the screen in mm and a trial index.
x1
The column name (between quotes, e.g. ‘x1’) or the number of the column in the dataframe containing the x-coordinates
y1
The column name (between quotes, e.g. ‘y1’) or the number of the column in the dataframe containing the y-coordinates
x2
When tracking was binocular, the column name (between quotes, e.g. ‘x2’) or number of the dataframe containing the x-coordinates of the second eye
y2
When tracking was binocular, the column name (between quotes, e.g. ‘y2’) or number of the dataframe containing the y-coordinates of the second eye
d1
The column name (between quotes, e.g. ‘d2’) or number of the dataframe containing the distance in mm
d2
When tracking was binocular, the column name (between quotes, e.g. ‘d2’) or number of the dataframe containing the distance in mm of the second eye
trial
The column name (between quotes, e.g. ‘TRIAL_INDEX’) or number of the dataframe containing the trial or stimuli index
height_px
The height of the stimuli in pixels, can be a single value or a vector of length number of trials when stimuli differ in size per trial
height_mm
The height of the stimuli in mm, can be a single value or a vector of length number of trials when stimuli differ in size per trial
width_px
The width of the stimuli in pixels, can be a single value or a vector of length number of trials when stimuli differ in size per trial
width_mm
The height of the stimuli in pixels, can be a single value or a vector of length number of trials when stimuli differ in size per trial
extra_var
A vector of names of the variables that must return in the output file, for example, condition, stimuli name, etc.
res_x
The horizontal resolution of the monitor in pixels
res_y
The vertical resolution of the monitor in pixels
samplerate
The samplerate, e.g., 60 for a 60 Hz scanner, 500 for a 500 Hz scanner
method
Must be one of the following methods: Mould, MouldDur, Mould.all, Mould.allDur, gazepath, dispersion, velocity.
Mould estimates velocity threshold per trial and duration thresholds per person (recommended for high, > 250 Hz, samplerate and high quality data)
MouldDur estimates velocity threshold per trial and uses the duration thresholds specified under ‘thres_dur’ (recommended for low, < 250 Hz, samplerate data and data of low quality)
Mould.all estimates velocity threshold and duration threshold for all trials (recommended for high, > 250 Hz, samplerate data and data with short trial times, < 2 seconds)
Mould.allDur estimates one velocity threshold for all trials and uses the duration threshold specified under ‘thres_dur’ (recommended for high, > 250 Hz, samplerate data and data with short trial times, < 2 seconds and few trials)
gazepath estimates velocity thresholds per person and uses the duration thresholds specified under 'thres_dur'. Furthermore, extra preprocessing steps are performed, such as, last point interpolation of missing data sequences of length in_thres and combining successive fixations that overlap in space.

dispersion derives fixations and saccades based on a dispersion threshold of .9 degrees of visual angle.

velocity classifies saccades as sequences of samples (> 4ms) that have a velocity over 35 deg/s. Fixations are the sequences of samples that are not saccades and pass the duration threshold of 100 ms.

posthoc TRUE or FALSE (default). When TRUE successive fixations close in space are merged.

thres_vel The desired velocity threshold, will only be used for method 'velocity'

thres_dur The desired duration threshold, will be used for methods 'gazepath', 'velocity' and 'MouldDur'

min_dist The minimal distance from the screen in mm, samples below this distance are considered unreliable and are omitted from further analysis.

in_thres The durations threshold in msec used by the gazepath method to interpolate.

object Object of class 'gazepath' can be used in the plot, summary and print function.

x Object of class 'gazepath' can be used in the plot, summary, head, tail and print function.

trial_index Argument of the plot function, refering to the trial to be plotted.

complete_only Argument (TRUE/FALSE) to indicate whether only fixations that have incoming out outgoing saccades and saccades that are between two fixations should be selected.

fixations_only Argument (TRUE/FALSE) to indicate whether only fixations should be returned by summary, instead of fixations and saccades.

Details

no details

Value

gazepath returns a list of 16 items. names(list) and be used to extract useful information, the plot() and summary() functions can be used to verify the procedure and obtain the fixations.

1. Classifications

List of length number of trials with a vector indicating whether a sample is classified as fixation 'f', saccade 's', unknown 'u' or NA.

2. x-coordinates

List of length number of trials with vectors of the x-coordinates of each trial.

3. y-coordinates

List of length number of trials with vectors of the y-coordinates of each trial.

4. Method

Single value indicating what method was used to classify fixations and saccades.
5. Robustness  Vector containing the mean durations of usable data segments per trial in milliseconds. Higher values indicate better data quality.

6. Precision  Vector containing the mean differences in pixels between raw data points and a smoothed version of the data. Higher values indicate poorer data quality.

7. Velocity thresholds  Vector with the velocity thresholds of each trial (methods: gazepath, Mould, MouldDur) in deg/s, or a single value when a single velocity threshold is used (methods: velocity, Mould.all, dispersion).

8. Duration threshold  Vector with the duration thresholds of each trial (method: Mould) in milliseconds, or a single value when a single duration threshold is used (methods: gazepath, MouldDur, Mould.all, velocity and dispersion).

9. Speed  List of length number of trials with vectors containing the speed at each sample.

10. Samplerate  The samplerate.

11. Head target distance  List of length number of trials with vectors containing the distance to the screen in mm for each sample.

12. Height in pixels  Vector of length number of trials with the stimuli heights in pixels.

13. Height in mm  Vector of length number of trials with the stimuli heights in mm.

14. Width in pixels  Vector of length number of trials with the stimuli widths in pixels.

15. Width in mm  Vector of length number of trials with the stimuli widths in mm.

16. Fixations and saccades per trial  List of length number of trials with dataframes containing the fixations and saccades, see also summary().

Author(s)
Daan van Renswoude & Ingmar Visser

References

Examples
```r
### Example
data(screen)
data(eye_dat)
example <- gazepath(eye_dat, x1 = 1, y1 = 2, d1 = 4, trial = 3,
height_px = screen[,1], height_mm = screen[,2],
width_px = screen[,3], width_mm = screen[,4],
method = 'gazepath', samplerate = 500)
```
## Use different i's to check the estimated fixations per trial
i <- 1
plot(example, trial_index = i)

## The estimated fixations, ordered per trial
s <- summary(example, complete_only = FALSE)

---

**GUI**

Graphical User Interface (GUI) for gazepath implemented in Shiny

**Description**

This function is a wrapper around gazepath and provides a graphical user interface (GUI) in shiny to parse raw eye-tracking data into fixations and saccades. The GUI functions opens a browser that serves as GUI to parse and visualize eye-tracking data.

**Usage**

GUI()

**Arguments**

no arguments

**Details**

no details

**Value**

nothing is returned

**Author(s)**

Daan van Renswoude & Ingmar Visser

**References**


**Examples**

```r
## Not run:
GUI()

## End(Not run)
```
Description
Screen and stimuli dimensions in pixels and mm, belonging to the dataset 'eye_data'.

Usage
data(screen)

Format
A dataframe with 4 variables (the height and width of stimuli in pixels and mm)

Source
S. P. Johnson

References
S. P. Johnson
Index

*Topic **GUI**
  GUI, 7

*Topic **datasets**
  eye_dat, 2
  screen, 8

*Topic **fixations**
  gazepath, 3
  GUI, 7

*Topic **gazepath**
  GUI, 7

*Topic **package**
  gazepath-package, 2

*Topic **saccades**
  gazepath, 3

eye_dat, 2

  gazepath-package (gazepath-package), 2
  gazepath, 3
  gazepath-package, 2
  GUI, 7

  head.gazepath (gazepath), 3

  plot.gazepath (gazepath), 3
  print.gazepath (gazepath), 3

  screen, 8
  summary.gazepath (gazepath), 3
  tail.gazepath (gazepath), 3